

Calibrating the pixel-level Kepler imaging data with a causal data-driven model

Dun Wang^{??}, Dan Foreman-Mackey^{??}, David W. Hogg^{??,??,??,??},
Bernhard Schölkopf^{??}, others

ABSTRACT

In general, astronomical observations are affected by several kinds of noise, each with its own causal source; there is photon noise, stochastic source variability, and residuals coming from imperfect calibration of the detector or telescope. In particular, the precision of NASA Kepler photometry for exoplanet science the most precise photometric measurements of stars ever made appears to be limited by unknown or untracked variations in spacecraft pointing and temperature, and unmodeled stellar variability. Here we present the Causal Pixel Model (CPM) for Kepler data, a data-driven model intended to capture variability but preserve transit signals. The CPM works at the pixel level (not the photometric measurement level); it can capture more fine-grained information about the variation of the spacecraft than is available in the pixel-summed aperture photometry. The basic idea is that CPM predicts each target pixel value from a large number of pixels of other stars sharing the instrument variabilities while not containing any information on possible transits at the target star. In addition, we use the target stars future and past (auto-regression). By appropriately separating the data into training and test sets, we ensure that information about any transit will be perfectly isolated from the fitting of the model. The method has four hyper-parameters (the number of predictor stars, the auto-regressive window size, and two L2-regularization amplitudes for model components), which we set by cross-validation. We determine a generic set of hyper-parameters that works well on most of the stars and apply the method to a corresponding set of target stars. For these stars, we calculate the Combined Differential Photometric Precision

¹Center for Cosmology and Particle Physics, Department of Physics, New York University

²Center for Data Science, New York University

³Max-Planck-Institut für Astronomie, Heidelberg, Germany

⁴To whom correspondence should be addressed; <david.hogg@nyu.edu>.

⁵Max-Planck-Institut für Intelligente Systeme, Tübingen

(CDPP), which indicate the noise level of a certain time scale. We find that we can consistently outperform (for the purposes of exoplanet detection) the Kepler Pre-search Data Conditioning (PDC) method for exoplanet discovery.

1. Ultra-precise photometry

The photometric measurements of stars made by the *Kepler* Satellite are precise enough to permit discovery of exoplanet transits with depths smaller than 10^{-4} . This precision results from great spacecraft stability, supplemented by various methods for removing small residual spacecraft-induced and stellar-variability trends in the brightnesses, either filtering the data (with things like median filters; **citations**) or fitting the data with flexible models (like polynomials or splines or Gaussian Processes; **citations**). When employed in the service of exoplanet search and characterization, these methods are usually agnostic about whether photometric variations originate in the spacecraft or in the star itself; that is, they obliterate intrinsic stellar variability along with spacecraft issues.

In general, there are many reasons for apparent photometric variability in a *Kepler* source. There is intrinsic stellar variability, which is of interest to some and a nuisance to others. There is also variability of overlapping fainter stars; that is, confusion noise combined with variability of the confusing sources. There are small changes in spacecraft pointing, which leads to slightly different illumination of the focal-plane pixels, and thus different sensitivity to errors (problems) in the device flat-field or sensitivity map. There are also *intra-pixel* sensitivity variations that can contribute (**cite white paper; Spitzer results**). There are small changes in spacecraft temperature, which lead to point-spread function (PSF) and differential (across the focal plane) pointing changes. These also lead to changes in pixel and intra-pixel illumination. Stellar proper motion, geometric parallaxes, and differential stellar aberration as the spacecraft orbits all do more of the same. There is electronic cross-talk between detectors and charge-transfer inefficiency; these can effectively transfer variability from one source to another. There are additional electronics effects like “rolling bands” that put additional features into lightcurves. There are also probably changes to the detector sensitivity with temperature and time, possibly illumination-history effects, and possibly sources of variability not yet considered. The remarkable thing about *Kepler* is that it is trying to measure stars at a level of precision much higher than ever previously attempted; new effects really *must* appear at some point. In Figure ??, we show the pixel-level variability in the *Kepler* data near one bright star that shows minimal intrinsic variability; this figure highlights the spacecraft-induced effects.

We propose and advise dealing with these variations in *Kepler* light-curves by *modeling*

them. In this context, a model is (at least) a parameterized function that can predict data, given parameter settings, and an objective function that can be optimized or sampled to choose or set those parameters. Ideally the objective function is probabilistic, or has a probabilistic justification, as with a likelihood or a posterior pdf. The model can be a physical model (of the spacecraft PSF, pointing, temperature, and so on) or it can be a flexible, effective model that has no direct interpretation in terms of physical spacecraft parameters. In principle a physical model will do a better job, because it necessarily embodies more prior information, but it requires research and intuition about dominant effects, and this research and intuition is often wrong or incomplete. The model we propose here is in the non-physical, effective category. The *Kepler* community is familiar with these kinds of models; to our knowledge, *all* successful light-curve “de-trending” methods are flexible, effective models.

One such method—one that is designed to describe or model spacecraft-induced problems but *not* interfere with measurements of stellar variability—is the *Kepler* Presearch (*sic*) Data Conditioning (PDC) method (**citation**). The PDC “de-trends” the *Kepler* lightcurves by fitting them with a small set of basis lightcurves generated from a principal components analysis (PCA) of filtered lightcurves. That is, it uses data to predict data, regularizing the fit (and avoiding over-fitting) by filtering and restricting the dimensionality (through PCA). The method proposed here, CPM, is very similar in spirit to the PDC. The main differences are (1) that CPM works in the pixel domain, not the lightcurve domain, so it has access to more fine-grained information, (2) that CPM has far more freedom (far more parameters) than the PDC but it strictly avoids over-fitting the lightcurves on exoplanet-transit time-scales through strong regularization and a train-and-test framework, and (3) that purpose of the CPM is to search for exoplanets, so in CPM, intrinsic stellar variability is also removed as well as the systematics. The two methods (CPM and PDC) are similar however, in that they both make the assumption that whatever spacecraft effects are imprinting variability on a stellar lightcurve must also be imprinting similar or related variability on other lightcurves. They also assume that the relationships can be captured with linear models. By going to the pixel level (unlike the PDC, which works at the photometry level), CPM makes it easier for the model to capture variability that is coming through variations in the centroids and point-spread function from spacecraft pointing, roll, and temperature.

Before we start, a few reminders about the *Kepler* data are in order: The satellite always observes precisely the same field, at fixed pointing (as closely as possible). Only a few percent of the 80 million pixels in the focal plane are telemetered down to Earth from each 30-min exposure; the telemetered pixels are associated with *Kepler* target stars chosen for study by the *Kepler* Team. The satellite is rolled by 90 deg every 90-ish days (to satisfy Solar-Angle constraints). The focal plane contains many CCDs; this plus the 90-deg rotations means that

each star is at a particular location on a particular CCD for 90-ish-day contiguous periods of time. The PSF varies strongly across the field and is badly sampled. The stars span a huge range in brightness; some of *Kepler*’s most important targets even saturate the device and bleed charge. The stellar photometry returned by the *Kepler* SAP and PDC pipelines is based on straight, *unweighted* sums of pixels in small patches centered on the stellar centroid. We will return to this latter point at the end; this kind of photometry cannot be optimal; it must be possible to do a better job with the photometric measurements. That’s beyond the scope of this project but a place for a valuable intervention on the *Kepler* data.

The scope of this project is an intervention—CPM—that makes the *Kepler* photometry more precise. We describe the method and deliver all the relevant code in a public, open-source repository. We also provide an interface to the *Kepler* data that delivers improved “CPM photometry” for every *Kepler* target.

2. Causal data-driven model

The first general idea about data-driven modeling of the kind used here is that each data point or data source is going to be *predicted* with or using a parameterized mathematical function of *other data*. That is, given a choice of some *target Kepler* data, we are going to find parameters of a function that takes as input some of the other *Kepler* data, and provides as output predictions of the target data. The simplest models are *linear models*, in which predictions for the target data are built from linear combinations of the other data, and in which the objective functions are quadratic in the prediction residual. Examples of quadratic objective functions include Gaussian likelihoods, mean-squared-error, and chi-square statistics. These models are simple, not just because they are easy to express and compute, but because optimization is convex: There is only one optimum for the objective function.

The point of these models is to be *flexible*, so the usual approach is to make the input data set very large, and the number of parameters large, often as large as—or larger than—the target data available. Such fits require *regularization* to break degeneracies and control ill-constrained parameters. These regularizations do for optimizations what priors do for posterior probability inferences; they express the desired behavior of the fit in the absence of data or along directions in parameter space in which the data are not constraining. General regularizations or priors will break the convexity of linear models; if convexity is to be maintained, these regularizations also have to be quadratic, or else one of a small number of other known forms (one of which is L1-regularization). Given these considerations, it makes sense to try to build our pixel-level model using a linear model with a quadratic objective

function and a quadratic regularization; this is what CPM will be.

The second general idea about data-driven modeling is the investigator’s beliefs about the causal processes that generate the data are crucial in restricting the kinds of data that can be used as input to the model. That is, different assumptions about the physical properties of the data and the data-taking system lead to different structures for the data-driven model. In the case of *Kepler* data, if we examine two arbitrary stars far away from each other on the same CCD as in Fig ?? (every grid in the plot is a pixel light curve time series), we can obviously see that the two stars that are hundreds pixel away from each other do have similar trends. If we believe that different stars in the *Kepler* field vary independently (that is, are not physically synchronized in any way) because of the distance between them, then the only reason that one star might show variability that is strongly *predictive* (useful for prediction) of another star’s variability is that both stars are being observed by the same device or spacecraft. That is, one star’s pixels can be used to predict another star’s pixels inasmuch as spacecraft issues imprint on both stars in related ways; they share a common cause—the systematics. For another example, if we think the spacecraft is being affected by processes that take place over time-scales longer than a single read-out (for example, thermal processes), then it would be sensible to model the data at time t using not just simultaneous data, but also data from a range of times around t . For another example, if an investigator doesn’t care about preserving stellar variability, and just wants to detect exoplanet transits (say), pixels from the target star can be used to predict pixels from the target star, provided they are at large-enough time lags that they don’t overlap (in time) the signals of interest. That is, if the model is designed to fit not just spacecraft variability but also intrinsic stellar variability, the predictive model will be permitted to use as input pixels that *do* overlap the target star. In what follows, we are going to use input data both from pixels of other stars and the target star pixels’ past and future. These models ought to remove both spacecraft variability and intrinsic stellar variability, which is optimal for exoplanet transit searching.

The third general idea is that we need to control for over-fitting. That is, once you have a flexible-enough model you can in principle fit *anything*, whether it was caused by the spacecraft, intrinsic stellar variability, or a transiting exoplanet. How do you prevent a very flexible model from taking small noise-induced fluctuations in all the input data and carefully combining them linearly into detailed models for every nook and cranny in the target data? In most projects in astrophysics, over-fitting is controlled for by limiting model freedom. The model is restricted in the number of parameters (as in “you can’t have more parameters than data points”) or by limiting the dimensionality (as in principal components analysis) or by applying strong priors (as with smoothness priors, that effectively reduce freedom without explicitly reducing the number of parameters). In each case, the restriction on model freedom is controlled by some *hyper-parameter* (such as the number of inputs, or number of principal

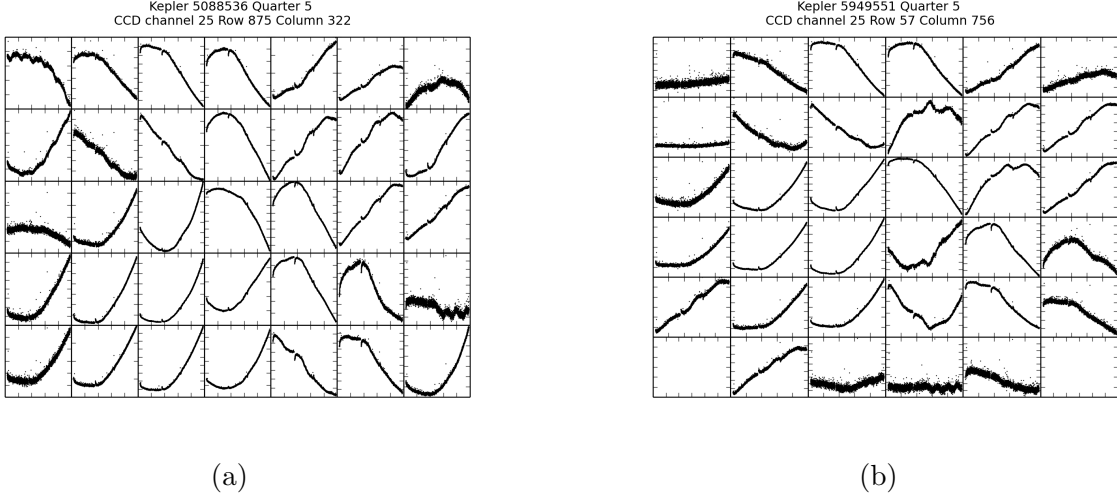


Fig. 1.—: Stars on the same CCD share systematic errors. The two panels show pixel fluxes (brightnesses) for two stars: (a) KIC 5088536, (b) KIC 5949551; here, KIC stands for Kepler Input Catalog. Both stars lie on the same CCD (see Fig. ??), but far enough apart such that there is no stray light from one affecting the other. Each panel shows the pixels contributing to the respective star. Note that there exist similar trends in these two stars, caused by systematic errors.

components, or the strength of the smoothness prior). The hyper-parameters can be set or tested with tools like cross-validation, the fully marginalized likelihood (Bayes factor or evidence), chi-squared statistical criteria, or intuition or heuristics. Here we take a different and more general approach, which is to use a train-and-test framework.

In this framework, the data used to *train* the model—meaning set the values of the parameters of the model—are disjoint from the *test* data—meaning the data that are being predicted (the target data). Because we are extremely concerned with detecting Earth-analog exoplanet transits, which take about ten hours, we adopt an extreme version of the train-and-test framework, in which the training data are always separated from the test data by at least ten hours. That is, when we are using the model to predict a particular pixel in the target data taken at time t , we use parameters obtained by an optimization that makes use of only data that comes either at times prior to $t - \Delta t$ or after $t + \Delta t$, where Δt is a tunable parameter but which we will set to roughly 12 h. This ensures that no information about any exoplanet transit itself can be entering into the prediction of the pixels contributing to the stellar photometry. In general, if there is a scientific goal of preserving intrinsic stellar variability, or transit signals, on time-scales of τ , the parameter optimization ought to be based on training data taken with a time-exclusion zone of half-width $\Delta t > \tau$.

In addition, each training data set within which we set the parameters (by optimization) has a finite total time extent $t_{\text{max}} \approx 30$ d. The train-and-test framework, which includes data out to time t_{max} but excludes data within Δt of the test data, effectively assumes that the signals worth preserving have time-scales less than Δt but don’t recur on time-scales shorter than t_{max} . Those assumptions are good for our purposes but not necessarily ideal for all users: Short-period exoplanets and certain kinds of stellar variability could be wiped out by a model with these settings of the training-data regime.

The fourth general idea involved in this kind of data-driven modeling is that the models aren’t *interpretable*. This means, in particular, that although the model might do a good job of *predicting* the target pixels, using a linear (or more complex) combination of the input pixel data, it won’t deliver anything that can be unambiguously interpreted as the *flux* of the star in question, or any other signal we care about. The data-driven model *effectively* describes the pointing, point-spread function, and flat-field of the telescope and camera, plus the variability of every star and every exoplanet transit, but it does so without ever *explicitly* creating any of those objects. We have to make some kind of heuristic or interpretive move to extract from the data-driven model the quantities of interest.

Finally, the fifth general idea is that there is no objective *ground truth* against which we can tell that any particular data-driven model is better or worse than any other. This problem is a problem for CPM, and for the *Kepler* PDC, and any other data-driven calibration models. One might think that the “best” model is the one that predicts data with lowest variance; this would be true if all models adhered to the same train-and-test framework, which they don’t. Besides, a model that can predict not just the spacecraft-induced variability, but also variability caused by exoplanet transits of interest, will effectively over-fit and distort the most important information in the light-curves. That is, what is considered best for modeling the data depends on the objectives of the user. For us, who are interested (in the long term) in finding and characterizing Earth analogs, the “best” data-driven model is the one that produces the most success in finding and characterizing Earth analogs! What we will show in what follows is that the CPM photometry does not distort artificial exoplanet transit signals injected into real *Kepler* pixel-level data. and promising properties of the CPM outputs. In the end, the value of the CPM will be demonstrated by the scientific projects it enables.

3. CPM specification and hyper-parameters

3.1. CPM model

Each individual *Kepler* target star is, for each month (here we always use data per month rather than quarter, since there is discontinuity between every 30-ish-day period), on a particular CCD on the *Kepler* focal plane. Associated with each target star is some set of pixels, located in a small contiguous patch more-or-less centered on the target star, and telemetered down once for every 30-min exposure. (Here we are ignoring all short-cadence data that is taken in a different mode with shorter exposures.) That is, each telemetered pixel in each CCD is associated with a particular *Kepler* target star.

Let’s consider a particular pixel m in the focal plane in one particular month. In that quarter, the spacecraft delivers $N \sim 1300$ measurements I_{mn} of the intensity falling in that pixel at the N times t_n at which exposures were taken during the month. (For intensity measurements I_{mn} here we are using pixels from the Target Pixel Files.) We want to build a prediction for the intensity I_{mn} in pixel m at each time t_n using the intensities $I_{m'n}$ of other pixels m' . The question is: What other pixels to choose? There are many possible qualitatively and quantitatively different choices here. Somewhat arbitrarily, we choose all the pixels m' associated with the Q stars on the same CCD in the same quarter that are closest in magnitude (*Kepler* magnitude as reported in the *Kepler* Input Catalog) to the target star associated with pixel m . This set of pixels—from the Q stars on the same CCD—is the *pixel set* \mathcal{M}_m associated with pixel m . To rule out any direct optical cross-talk by stray light, we require that the predictor pixels are from stars sufficiently far away from the target star (at least 20 pixels distance on the CCD). Note that because the set \mathcal{M}_m is of pixels associated with *different* stars than the star associated with pixel m , and are far from it on CCD, the pixel m will not be in the set \mathcal{M}_m , and nor will any of pixel m ’s close neighbors. That is, there will be no (or almost no) overlap in stellar illumination of pixel m and the pixels $m' \in \mathcal{M}_m$.

When we are predicting the measurement I_{mn} from pixel m at a particular time t_n , we are going to use a train-and-test framework in which not only do we not use data at time t_n in optimizing the parameters of the regression or fit, but we don’t even use any times t such that $|t - t_n| < \Delta t$, where $\Delta t \sim 10$ h as shown in Fig. ?? . That is, we train (optimize fit parameters) using the *time set* \mathcal{N}_n of time indices n' such that for all $n' \in \mathcal{N}_n$, $t_{n'}$ is in the same quarter as t_n , and $|t_{n'} - t_n| > \Delta t$. The time set of indices \mathcal{N}_n therefore does not overlap the time point t_n , nor any of its neighbors in a time window of half-width Δt .

In addition to pixel set $m' \in \mathcal{M}_m$ from other stars, we also include past and future of the target star — autoregressive (AR) component as input to remove more of the stellar

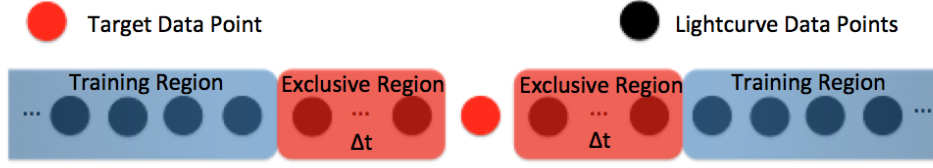


Fig. 2.—: Train-and-test framework. Data near the target point t_n within Δt is excluded in the training set

variability and thus increase the sensitivity for transits. To do this, we select an exclusion window of half-width Δt including R data points, where $\Delta t \sim 10$ (as big as the window in train-and-test framework) around the point of time t_n being corrected, to ensure that we do not remove the transit itself and then use S closest future and past time points. That is, for every pixel m in the target star, we construct S virtual time series, in which every component v is defined to be

$$I_{vn} = I_{mn-R-k} \text{ or } I_{vn} = I_{mn+R+k}, \quad , \quad (1)$$

where $k = 0, 1, \dots, S$, so if there are totally M pixels in the target stars, we will finally add $2 \cdot M \cdot S$ autoregressive components into the predictors, which construct the autoregressive set \mathcal{V}_m

As mentioned above, in the context of this project, a model is something that can predict data given settings of some parameters, and an objective function that can be used to find best values for those parameters. We prefer to work in a probabilistic mode in which the objective function can be justified in terms of a likelihood (probability for the data given the model) or a posterior probability density function (likelihood times a prior pdf). The CPM treats each *Kepler* data point I_{mn} as being predictable from a linear combination of data points $I_{m'n}$, where m' is from the set of non-overlapping pixels $m' \in \mathcal{M}_m$.

$$I_{mn} = I_{mn}^* + e_{mn} \quad (2)$$

$$I_{mn}^* = \sum_{m' \in \mathcal{M}_m} a_{mnm'} I_{m'n} + \sum_{v \in \mathcal{V}_m} b_{mnv} I_{vn} \quad , \quad (3)$$

where I_{mn}^* is the prediction (by the model) for data point I_{mn} , e_{mn} is a noise contribution or residual away from the prediction, and the $a_{mnm'}$ and b_{mnk} are the parameters (linear coefficients of the prediction). The parameters $a_{mnm'}$ and b_{mnk} have indices m because they are different for every pixel m , and they will even be different for every time step n ; that is, there will be a separate best-fit value for the parameters for every I_{mn} . If we presume that the residuals away from the prediction are normally distributed with zero

mean and known variance, likelihood optimization reduces to chi-squared minimization. We add to the standard chi-squared definition a regularization term (equivalent to multiplying the likelihood by a Gaussian prior pdf) that penalizes large absolute values for the coefficients $a_{mnm'}$:

$$\chi_{mn}^2 = \sum_{n' \in \mathcal{N}_n} \frac{[I_{mn'} - I_{mnn'}^*]^2}{\sigma_{mn'}^2} + \lambda_a \sum_{m' \in \mathcal{M}_m} a_{mnm'}^2 + \lambda_b \sum_{v \in \mathcal{V}_m} b_{mnv}^2 \quad (4)$$

$$I_{mnn'}^* = \sum_{m' \in \mathcal{M}_m} a_{mnm'} I_{m'n'} + \sum_{v \in \mathcal{V}_m} b_{mnv} I_{vn'} \quad , \quad (5)$$

where the $\sigma_{mn'}^2$ are the (presumed known) individual-pixel noise variances, and λ_a, λ_b set the strength of the regularization (or width of the prior pdf) for parameters $a_{mnm'}, b_{mnv}$.

The train-and-test idea comes into the objective function χ_{mn}^2 : When we are computing the objective function χ_{mn}^2 , we are using only time points $n' \in \mathcal{N}_n$ that don't overlap the target (or test) time point n . We train the model—that is, set the parameters $a_{mnm'}$ and b_{mnv} —by choosing the full set of parameters that jointly minimize the objective function χ_{mn}^2 . Fortunately, given the form of the model, this minimization is just a linear solve (linear operation on the data). Importantly however, and perhaps surprisingly, the objective function χ_{mn}^2 is *different* for every target data point (pixel value to be predicted) I_{mn} ; we have to do an *independent optimization* (or really linear solve) of the parameters for every pixel value we want to predict. That is, every pixel datum I_{mn} we predict will have *different* settings of the parameters $a_{mnm'}$ and b_{mnv} . Since there are of order 10^6 total pixel values in the *Kepler* Target Pixel Files for *each Kepler target*, and of order 10^7 data sources used as basis functions in the linear fitting, this represents one fuck of a lot of linear solves, and a lot of parameters to obtain and record.

Once we have obtained the parameters $a_{mnm'}$ and b_{mnv} corresponding to a particular data point I_{mn} , we can make a *prediction* I_{mn}^* for the intensity at that pixel. By construction of our objective function, this is truly a prediction, in the sense that the optimization (or solve) that produced the parameters (and thus the prediction) did not make use of I_{mn} itself, or even any of the pixel values that are nearby in angle or time (as described above). With all the predicted pixel values I_{mn}^* of the target star, our new stellar flux estimates, what we will call the “CPM prediction”, is constructed from the CPM pixels in precisely the same way as the official *Kepler* SAP photometry is constructed from the Target Pixel Files. That is, we perform a weighted sum of CPM predicted pixels I_{mn}^* with weights w_m , all of which are either zero or unity, and we adopt precisely the same weight assignments as are adopted in the SAP photometry; in equations, the flux estimate S_n at time t_n of the target star is

given by

$$S_n = \sum_m w_m I_{mn}^* \quad , \quad (6)$$

where the sum is over the M pixels that are associated with the target star. As we have noted above and below, these photometric estimators are not optimal for any purpose, but improving them is beyond the scope of this project.

It is worth to mention that the CPM prediction S_n is constructed from pixels without any transit information. Therefore, it can be regarded as a light curve without signal. Based on this idea, we can construct the “CPM flux” to be the relative residual between CPM prediction and SAP Flux F_n

$$\delta_n \equiv \frac{F_n - S_n}{S_n} \quad . \quad (7)$$

The CPM flux is In some sense, it is this relative residual that contains the exoplanet transit signals we seek. That is, a transit creates a negative residual away from the prediction.

3.2. Hyper-parameters

In the description of CPM, we introduce 4 hyper-parameters—number of predictor stars Q (or number of predictor pixels P), number of autoregressive components S , and the two regularization strength λ_a and λ_b . To set these 4 hyper-parameters, in principle we need to run cross-validation on every single stars to optimize the performance of CPM. But optimizing in this 4-dimensional space is expensive, especially in CPM we need to solve thousands of linear system with thousands of parameters. Therefore, in this paper we just set a general set of hyper-parameters ($P = 4000$, $Q = 3$, $\lambda_a = 1e5$, $\lambda_b = 1e5$) which works well in most of the stars without highly variability. To show how this general set of hyper-parameters performs, here we present a typical comparison between optimized and default hyper-parameters in Fig ?? . We can see that, in Fig ??, there is less variation in the CPM flux from optimized hyper-parameters (green points in second panel) than from the default one (red points in the bottom panel). In addition, the optimized flux has higher signal to noise ratio of the known transit than the default. However, the performance of default hyper-parameters is acceptable and we will use this set of hyper-parameters throughout rest of the paper to show the general performance of CPM.

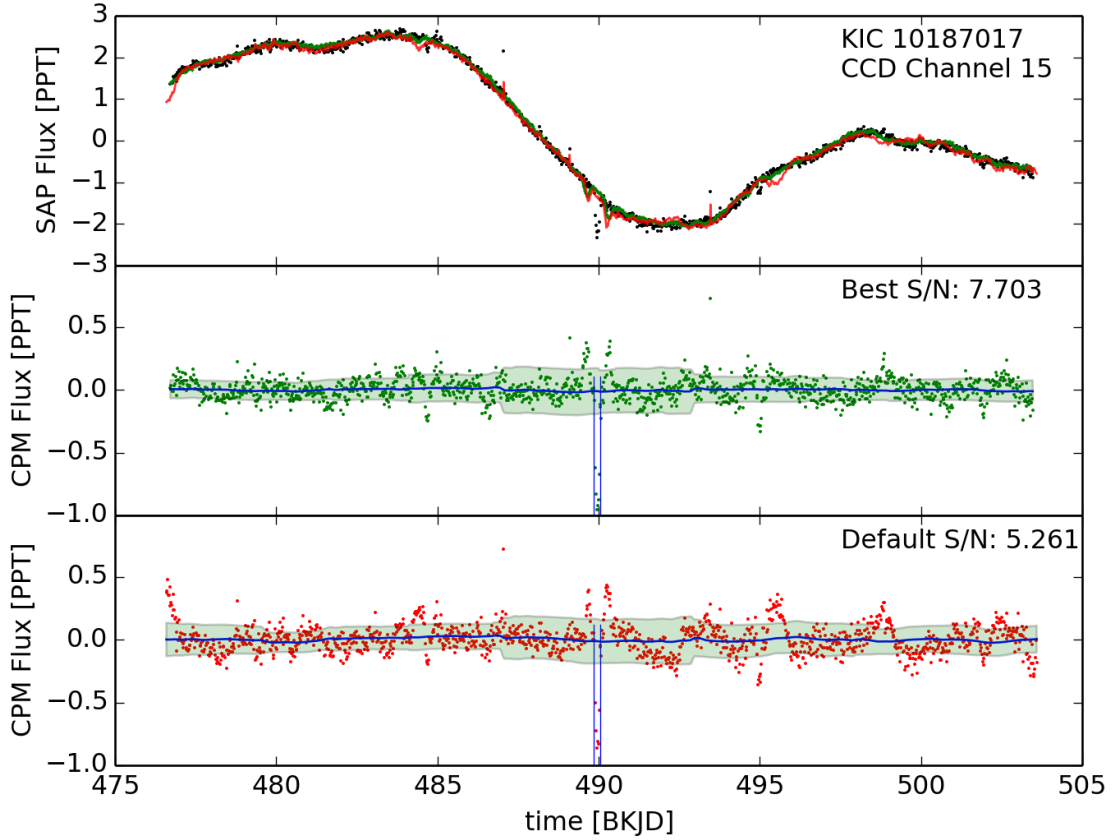


Fig. 3.—: Comparison between default and optimized hyper-parameters. The three panels plot show the results from default, In the top panel, SAP flux is plotted in black, CPM prediction of default hyper parameters is in red line, CPM prediction of optimized hyper-parameters is in green line. The middle panel shows the CPM flux of the optimized hyper-parameters, while the bottom is the CPM flux of the default hyper-parameters, the two vertical blue lines indicate the location of the transit signal, and signal to noise ratio is calculated in both situation. The optimized hyper-parameter performs better than the Default one, but the results from both situation are close.

3.3. Transit signal in CPM

One important feature of CPM is that we use the train-and-test framework to preserve the transit signal. To show how well train-and-test framework performs, we make the following experiments on *Kepler* data as shown in Fig ?? In the light curve of single pixel (1, 2) of KIC 9822284, a 10-hrs fake signal with amplitude 1.0005 is inserted. To insert the signal, we simply multiply the light curve by the factor of 1.0005, which makes the little bump in the middle of the SAP light curve (black points in the first panel). With the distorted light curve, we both applied the CPM and ordinary fitting to the data. In the result, we can see that both CPM and ordinary fit the light curve quite well outside the range of inserted signal. However, when examine how these two models preserve the signal that we care, the ordinary fitting fit out about 30 percent of the signal, while CPM almost preserve the whole original amplitude. Therefore, by this simple experiment, we are able to show that the train-and-test framework used here is capable to keep the transit signal with a particular time scale. And we want to emphasise that it is crucial for searching for exoplanets, especially the small ones such as earth-like planets, which only have about tens of parts per million signal in the light curve.

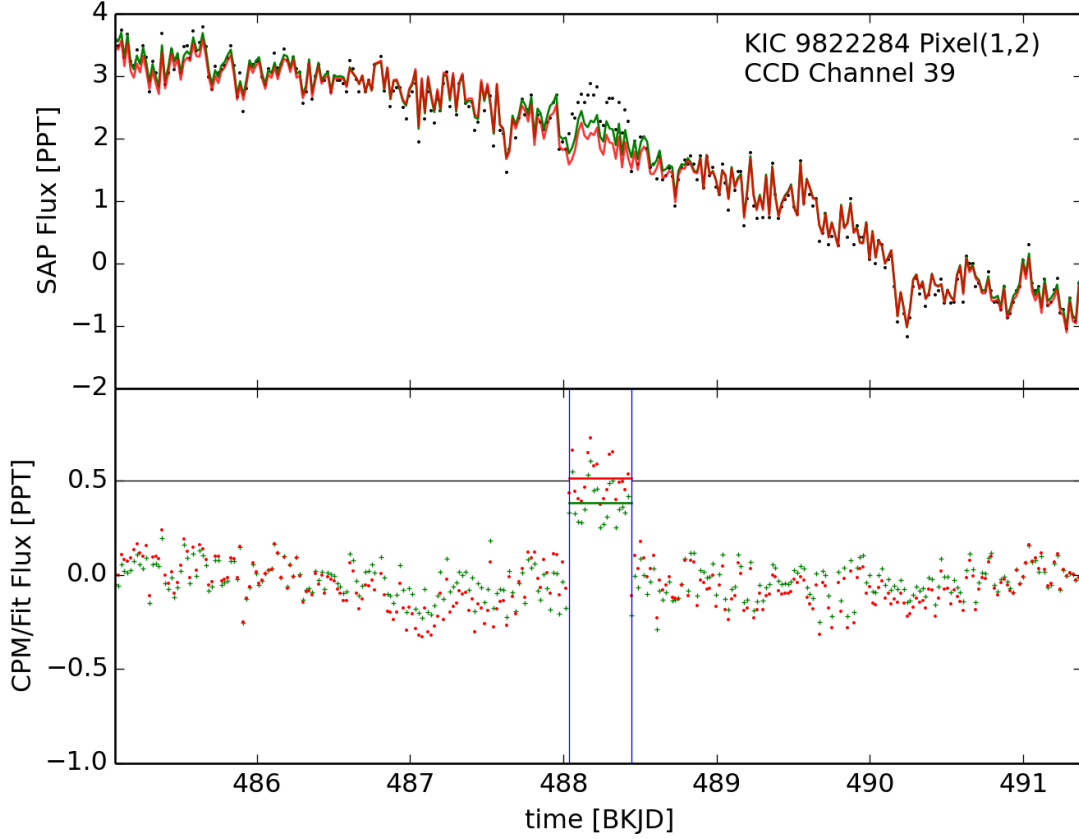


Fig. 4.— CPM preserves transit signals. Fake signal with amplitude 1.0005 is inserted into the single pixel (1, 2) of star KIC 9822284. In the top panel, the black points shows the SAP flux of the pixel with the injected signal, the green line is the fitting of the SAP flux, and the red line is the CPM prediction. In the bottom panel, red and green points represent the CPM flux and fit flux, while the signal value is indicated by the horizontal line. CPM largely preserves the signal while the fitting flux overfit the signal.

4. Examples and results

To give a view on how our method performs, CPM is applied on several stars with known transit signals with different variability and magnitudes as shown in Fig. ?? . From the plots, we can see that CPM works quite well on these stars, producing the CPM fluxes with low variation and preserving the transit signals. In addition, we compare 6 quiet stars (non-variable stars) with the PDC method. The results illustrate that for these quiet stars our approach removes a major part of the variability present in the PDC light curves, while preserving the transit signals. To provide a quantitative comparison, we ran CPM on 1000 stars from the whole Kepler input catalog (500 chosen randomly from the whole list, and 500 random G-type sun-like stars), and estimate the Combined Differential Photometric Precision (CDPP) for CPM and PDC (before CDPP is estimated, the 100 most variable stars in the list are removed, since PDC is not designed to remove stellar variability). CDPP is an estimate of the relative precision in a time window, indicating the noise level seen by a transit signal with a given duration. The duration is typically chosen to be 3, 6, or 12 hours. We use the 12-hours CDPP metric, since the transit duration of an earth-like planet is roughly 10 hours. Fig. ?? presents our CDPP comparison of CPM and PDC, showing that our method (CPM) outperforms PDC.

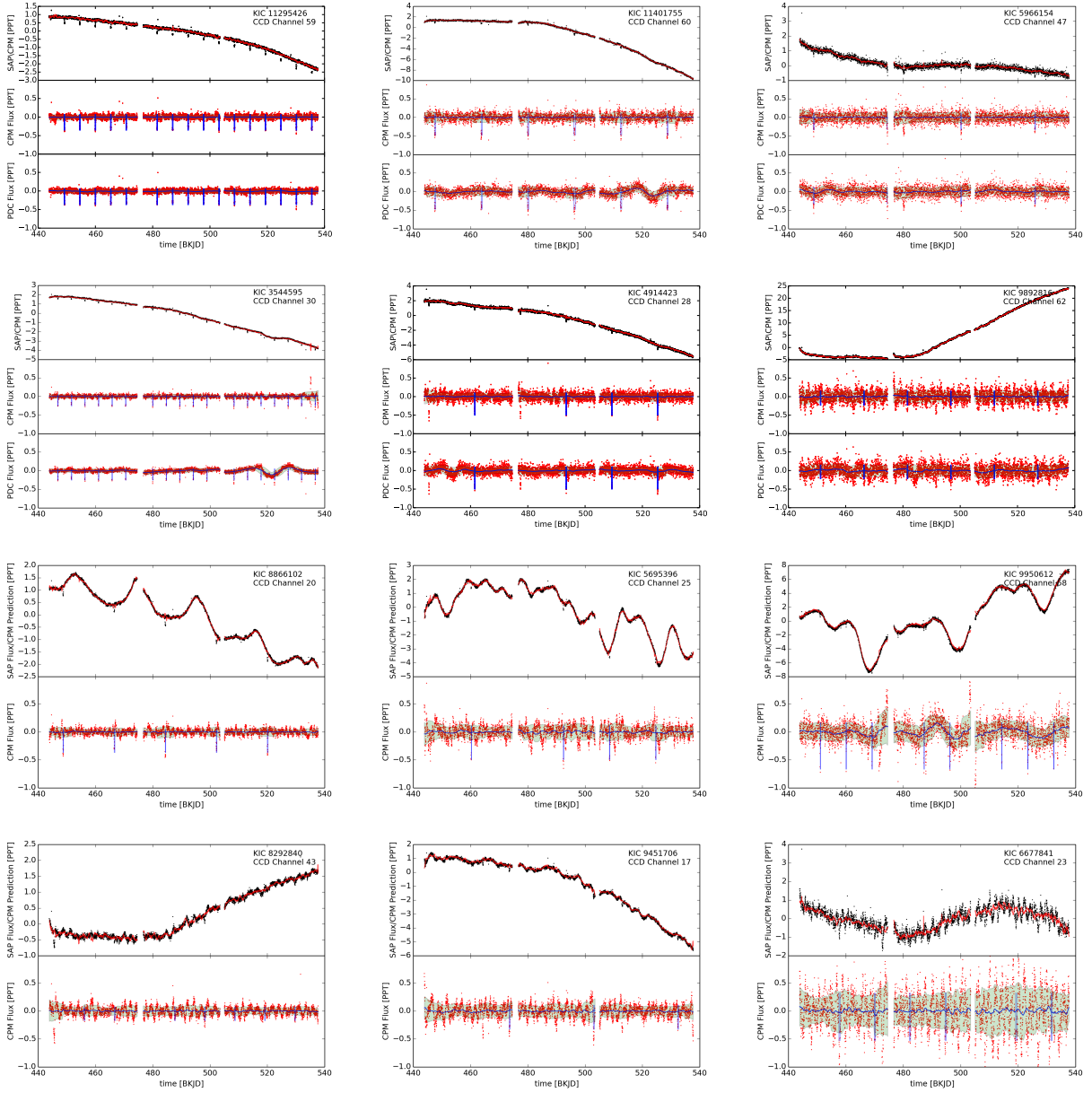


Fig. 5.—: Corrected fluxes using CPM, for 12 example stars, spanning the main magnitude (brightness) range encountered. In first column plots, we consider bright stars with magnitude from 9 to 10.5, second column are stars with magnitude from 11 to 12.5, and the third column are faint stars with magnitude above 13. In the first two rows, we present some quiet stars. In all three panels, the top panel shows the SAP flux (black) and the CPM regression (red), i.e., our prediction of the star from other stars. The middle panel shows the CPM flux corrected using the regression, and the bottom shows the PDC flux. One can see that the CPM flux curve preserves the exoplanet transits (little downward spikes), while removing a substantial part of the variability present in the PDC flux. Both the bottom two rows present some variable stars with slow and fast variability. For the variable stars, we did not present the PDC comparison, since the PDC flux try to preserve intrinsic stellar variability.

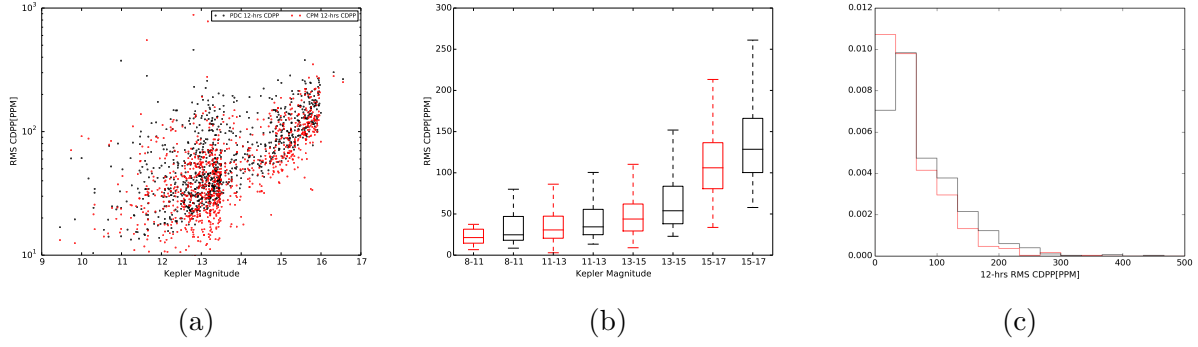


Fig. 6.— Comparison of the proposed method (CPM) to the Kepler Pre-search Data Conditioning (PDC) method in terms of Combined Differential Photometric Precision (CDPP) (see text). Plot (a) shows our performance (red) vs. the PDC performance in a scatter plot, as a function of star magnitude (note that larger magnitude means fainter stars, and smaller values of CDPP indicate a higher quality as measured by CDPP). Plot (b) bins the same dataset and shows box plots within each bin, indicating median, top quartile and bottom quartile. The red box corresponds to CPM, while the black box refers to PDC. Plot (c), finally, shows a histogram of CDPP values. Note that the red histogram has more mass towards the left, i.e., smaller values of CDPP, indicating that our method overall outperforms PDC, the Kepler “gold standard.”

5. Discussion

We have presented a simple yet highly effective method that works in pixel-level based on the causal structure of the *Kepler* data to calibrate the *Kepler* light curves, which is intended to search exoplanets. However, based on our assumption of the causal structure of the *Kepler* data, if we turn off the auto-regressive components and keep the predictors from other stars working in the CPM, we should be able to still remove the systematics while preserve the intrinsic stellar variability, since there is no reason that these independent stars can be used to predict the stellar variability of the target star. However, in fact, we can not preserve the stellar variability just by turning off the auto-regressive components. One possible reason can be that, since our model is so flexible (thousands of parameters), in principle it can fit any functions. In this sense, we should run cross-validation with the auto-regressive components turned off and to find out how many pixels from other stars we need to just remove systematics while preserve stellar variability, but in fact we do not have a good criterion to check how well we preserve stellar variability in cross-validation because we do not have enough knowledge of the true stellar variability. Thus for the time being, we just focus on producing well-calibrated light curves for searching exoplanets, but in long term, we are looking forward to find a method to preserve the stellar variability.

However, forget all the unsolved problems, we expect that our method will enable astronomical discoveries at higher sensitivity on the existing *Kepler* data as well as on future missions. As we know, *Kepler*K2 data are suffering from the broken-down of reaction wheels, which makes the satellite hard to control and introduce significant systematics. Moreover, in 2017, NASA is planning the launch of another space telescope—TESS (Transiting Exoplanet Survey Satellite), which will perform an all-sky survey for small (earth-like) planets of nearby M stars. We think our method can be simply extended for these project to help achieving higher precision and more scientific results.

It is a pleasure to thank the whole *Kepler* Team for designing, delivering, and operating a great facility, and for making all of the data public, in all its rawest forms, through the MAST interface. We are also pleased to thank Ruth Angus (Oxford), Tom Barclay (Ames), Bekki Dawson (Berkeley), Rob Fergus (NYU), Stefan Harmeling (Dusseldorf), Michael Hirsch (UCL), Dustin Lang (CMU), Benjamin T. Montet (Caltech), David Schiminovich (Columbia), and Jake Vanderplas (UW) for valuable discussions, input, encouragement, and advice. This project was partially supported by NSF grant IIS-1124794, NASA grant NNX12AI50G, the Moore Foundation, and the Sloan Foundation. This research made use of the NASA Astrophysics Data System.